

Porting KWin to Maemo/MeeGo

KWin Mobile

Martin Gräßlin
kde@martin-graesslin.com

Akademy 2010, Tampere

03.07.2010





Outline

- 1 Benefits of Mobile Edition
- 2 KWin Compositing
- 3 OpenGL and ES
- 4 Roadmap





Outline

- 1 Benefits of Mobile Edition
- 2 KWin Compositing
- 3 OpenGL and ES
- 4 Roadmap



Why porting KWin?

We want to...

- have more than one running app
- use known and stable technology
- easily trigger effects from Plasma shell
- reuse as much code as possible on different platforms
- optimize KWin compositing



State of Mobile Window Managers

Clutter based

- Mutter (Moblin)
- Matchbox (Maemo)

Future?

Which one will be used in MeeGo?

Single vendor lock-in

We need competition on mobile window management!



But KWin is so heavyweight...

Few dependencies

- X: Xlib, XComposite, XDamage, XRender, XRandr, XFixes
- OpenGL
- Qt
- KDE: KDE-UI, Plasma, kworkspace, kephal

Code Base consists of plugins

- Overall: 100 kSLOC
- Effects: 23 kSLOC
- Decos: 22 kSLOC
- KCMs: 10 kSLOC

For comparison

- Matchbox (Fremantle): 17 kSLOC
- Mutter (git master): 60 kSLOC
- Clutter (Fremantle): 57 kSLOC





Decoration Plugins

Decorationless Windows

- All windows open maximized
- No Taskbar (minimize)
- Close button in panel
- Decorations focus on Desktop systems

Aurorae

- For modal dialogs
- Shares styling system with Plasma
- Provides matching shadows





Decoration Plugins

Decorationless Windows

- All windows open maximized
- No Taskbar (minimize)
- Close button in panel
- Decorations focus on Desktop systems

Aurorae

- For modal dialogs
- Shares styling system with Plasma
- Provides matching shadows



Most Effects Are Not Needed

- boxswitch
- desktopgrid
- dialogparent
- diminactive
- dimscreen
- fade
- fadedesktop
- fallapart
- highlightwindow
- login
- logout
- magicclamp
- translucency
- minimizeanimation
- presentwindows
- resize
- scalein
- shadow
- showfps
- showpaint
- slide
- slideback
- slidingpopups
- taskbarthumbnail
- thumbnailside
- zoom
- blur
- coverswitch
- cube
- explosion
- flipswitch
- glide
- invert
- lookingglass
- magnifier
- mousemark
- sharpen
- sheet
- snaphelper
- snow
- trackmouse
- wobblywindows



Most Effects Are Not Needed

- boxswitch
- desktopgrid
- **dialogparent**
- diminactive
- dimscreen
- fade
- fadedesktop
- fallapart
- highlightwindow
- login
- **logout**
- magicclamp
- translucency
- minimizeanimation
- **presentwindows**
- resize
- scalein
- shadow
- **showfps**
- **showpaint**
- slide
- slideback
- **slidingpopups**
- taskbarthumbnail
- thumbnailaside
- zoom
- blur
- coverswitch
- cube
- explosion
- flipswitch
- glide
- invert
- lookingglass
- magnifier
- mousemark
- sharpen
- sheet
- snaphelper
- snow
- trackmouse
- wobblywindows



KWin Provides Mobile Functionality

Netbook Shell

- Windows open maximized
- Present Windows for switching
- Good interaction with Plasma Shell
- Working Focus Stealing Prevention

GSoC Project

Scripting Interface for KWin



KWin Provides Mobile Functionality

Netbook Shell

- Windows open maximized
- Present Windows for switching
- Good interaction with Plasma Shell
- Working Focus Stealing Prevention

GSoC Project

Scripting Interface for KWin





Outline

- 1 Benefits of Mobile Edition
- 2 KWin Compositing
- 3 OpenGL and ES
- 4 Roadmap



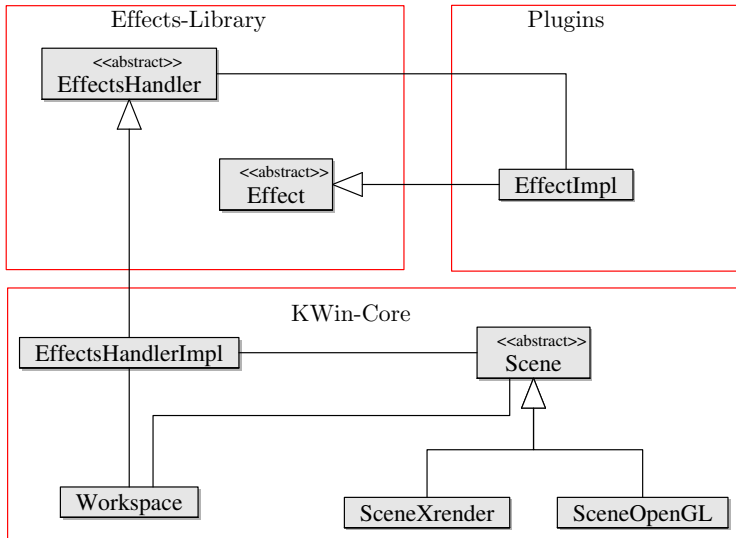
KWin Supports Multiple Backends

XRender and OpenGL

- XRender backend
- OpenGL backend
- Modular builds
- Abstraction layer for effects
- Effects can say what they need
- Effects can use OpenGL and XRender directly
- Plasma styled textures



Separation of Functionality





Integrating OpenGL ES

Possible Approaches

- 1 Merge OpenGL and ES code
- 2 Write New SceneOpenGLES
- 3 Branch and convert all OpenGL code to OpenGL ES

Problems

- 1 SceneOpenGL assumes glX
- 2 Duplicates all OpenGL code
- 3 Unmaintainable





Integrating OpenGL ES

Possible Approaches

- 1 Merge OpenGL and ES code
- 2 Write New SceneOpenGLES
- 3 Branch and convert all OpenGL code to OpenGL ES

Problems

- 1 SceneOpenGL assumes glX
- 2 Duplicates all OpenGL code
- 3 Unmaintainable





Outline

- 1 Benefits of Mobile Edition
- 2 KWin Compositing
- 3 OpenGL and ES**
- 4 Roadmap

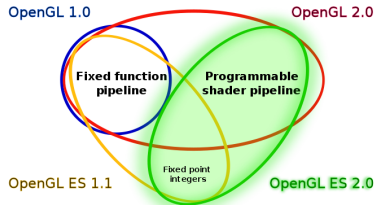




OpenGL is a Version Jungle

Fixed Functionality vs Programmable Pipeline

- 1.x uses fixed functionality
- 2.x introduces shaders
- 3.x deprecates fixed functionality
- ES 1.1 based on OpenGL 1.5
- ES 2.0 based on OpenGL 2.0



<http://wiki.maemo.org/OpenGL-ES>

Problem

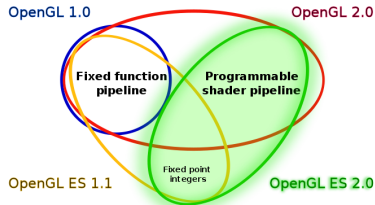
ES 2.0 is not backward compatible



OpenGL is a Version Jungle

Fixed Functionality vs Programmable Pipeline

- 1.x uses fixed functionality
- 2.x introduces shaders
- 3.x deprecates fixed functionality
- ES 1.1 based on OpenGL 1.5
- ES 2.0 based on OpenGL 2.0



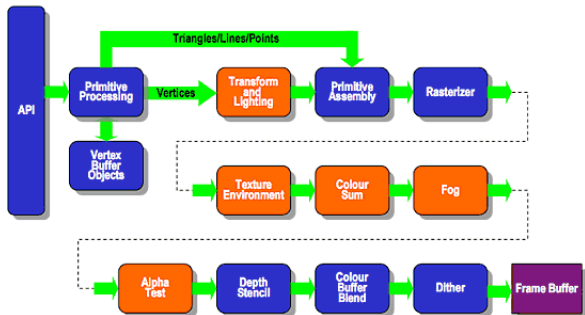
<http://wiki.maemo.org/OpenGL-ES>

Problem

ES 2.0 is not backward compatible



Existing Fixed Function Pipeline

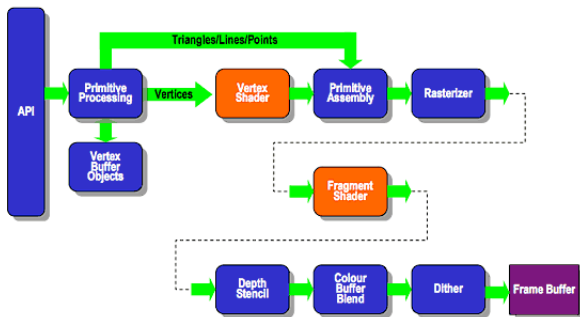


© Khronos, <http://www.khronos.org/opengles/2.X/>



Programmable Pipeline

ES2.0 Programmable Pipeline



© Khronos, http://www.khronos.org/opengles/2_X/



KWin uses Fixed Functionality

KWin's OpenGL

- KWin based on OpenGL 1.x
- Mostly old and deprecated API calls
- Some Shader based effects (OpenGL 2.x)
- Shader emulate fixed functionality

OpenGL ES on N900

- OpenGL ES 1.1
- OpenGL ES 2.0



KWin uses Fixed Functionality

KWin's OpenGL

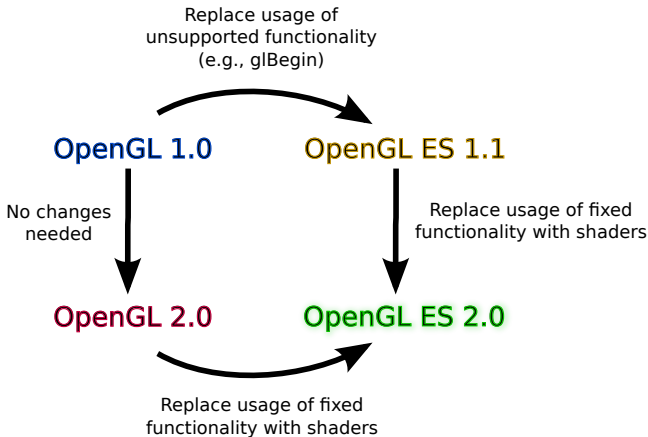
- KWin based on OpenGL 1.x
- Mostly old and deprecated API calls
- Some Shader based effects (OpenGL 2.x)
- Shader emulate fixed functionality

OpenGL ES on N900

- OpenGL ES 1.1
- OpenGL ES 2.0



Porting Between OpenGL Versions



<http://wiki.maemo.org/OpenGL-ES>



Deprecated Rendering Code

```
1 glBegin( GL_QUADS );
2 // This sucks. But makes it faster.
3 if( texture && color )
4 {
5     for( int i = 0; i < count; i++ )
6     {
7         glTexCoord2fv( texture + i*tsize );
8         glColor4fv( color + i*csize );
9         glVertexFunc( vertices + i*vsize );
10    }
11 }
12 else if( texture )
13 {
14     for( int i = 0; i < count; i++ )
15     {
16         glTexCoord2fv( texture + i*tsize );
17         glVertexFunc( vertices + i*vsize );
18     }
19 }
20 else if( color )
21 {
22     for( int i = 0; i < count; i++ )
23     {
24         glColor4fv( color + i*csize );
25         glVertexFunc( vertices + i*vsize );
26     }
27 }
28 else
29 {
30     for( int i = 0; i < count; i++ )
31         glVertexFunc( vertices + i*vsize );
32 }
33 glEnd();
```





Deprecated Rendering Code

Deprecated Code in ES 1.1

- glBegin/glEnd
- GL_QUADS
- glTexCoord
- glVertex
- glColor (in ES 2.0)

Problematic Code

- Usage of Quads





Deprecated Rendering Code

Deprecated Code in ES 1.1

- glBegin/glEnd
- GL_QUADS
- glTexCoord
- glVertex
- glColor (in ES 2.0)

Problematic Code

- Usage of Quads





Outline

- 1 Benefits of Mobile Edition
- 2 KWin Compositing
- 3 OpenGL and ES
- 4 Roadmap**





Current State

No code written

- Stabilizing 4.5 is more important
- Waiting for git

What is done?

- Experimental Port to OpenGL 3
- SceneOpenGL is analyzed for feasibility to port to ES





How To Go From Here

Schedule

- Remove deprecated OpenGL code
- Remove unused code in Scene and Effects Lib
- Drop XGL support
- Port to OpenGL ES 1.1 till 4.6
- Port to OpenGL 3 and ES 2.0 till 4.7





Questions?

